Human Identification Through Insecure Channel

Tsutomu MATSUMOTO

Hideki IMAI

Division of Electrical and Computer Engineering

YOKOHAMA NATIONAL UNIVERSITY

156 Tokiwadai, Hodogaya, Yokohama 240, Japan

Fax: +81-45-338-1157, Tel: +81-45-335-1451

Internet: tsutomu@mlab.dnj.ynu.ac.jp

Abstract This paper examines a relatively new problem of how to securely identify a human through an insecure channel. It proposes a simple but powerful cryptographic scheme that fits human ability of memorizing and processing. Typical applications of the scheme are the identification verification of an user at an on-line terminal of a central computer or holder verification done by an IC card which can communicate its holder only through an equipment like an automatic vendor machine.

Keywords Identification Verification, Human Identification, User Identification, Authentication, Cryptography, Access Control, Password, IC Card, Smart Card, Insecure Channel, Human Interface

1 Introduction

Human identification, or user identification, is one of the most important items for information security. Based on biometrics (eg., fingerprints), something memorized (eg., passwords), belongings (eg., tokens), and their combinations, a variety of human identification schemes have been developed and utilized actually. A compact and excellent survey appears in [1]. This paper proposes a scheme to securely identify a human through an insecure channel.

A popular human identification scheme is that a verifier V firstly requires a human prover P to simply exhibit a password and then accepts P if and only if the received password coincides with its registered counterpart. This scheme is very convenient and often used. However, for an attacker who can watch the interaction between a verifier and an accepted prover, it is an easy task to obtain the password between them and to masquerade as the prover as long as the same password is being used. Consider the following cases to make clear the crucial points.

Case 1: Suppose there is a terminal connected to a central computer V through a communication network. Let's call the path to V from a human P via the terminal

D.W. Davies (Ed.): Advances in Cryptology - EUROCRYPT '91, LNCS 547, pp. 409-421, 1991.
© Springer-Verlag Berlin Heidelberg 1991

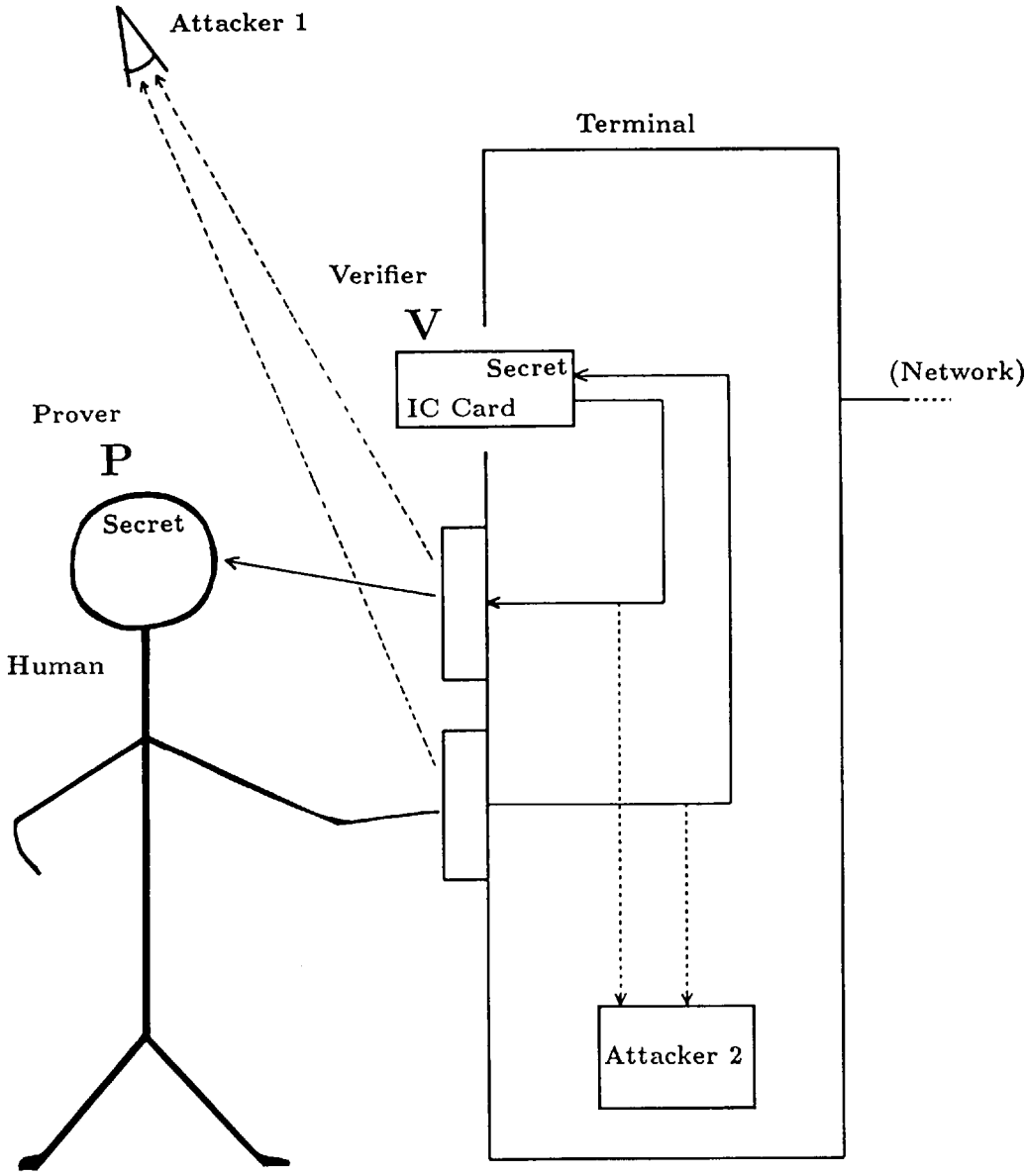

Figure 1. Insecure Channel

as the channel C. When human P requests an access to computer V, how should V verify the identity of P or how should P prove its identity to V by an interaction through channel C ? The line between the terminal and V can be protected by encipherment. However, it is hard to expect that channel C is always secure since how P operates the terminal may be watched by somebody standing behind P or some subliminal equipment set inside the terminal or everywhere in the network. As well as watching, the equipment might also behave as a true verifier to actively steal the secret of P.

Case 2: Imagine a human P keeping her/his own IC card (smart card) V with no direct human interfaces (i.e., no built-in keyboard and display). IC card V may be used for identification by belongings based on highly secure cryptographic techniques, which include brand-new zero-knowledge identification schemes. Suppose a system in which if P wants to use IC card V then P should connect V to a device C with IC card interface and direct human interfaces and P should give appropriate commands to C. Some automatic vendor machines and some automatic teller machines are existing examples of such C. Assume that IC card V is made to correctly work only when it has succeeded in verifying operator P as the true holder through device C. Thus C is the unique channel for communication between P and V. But it is hard to expect that channel C is always secure for P and V. The reason is that all the data transferred through C can be watched by somebody standing behind P or a subliminal equipment set inside C which may also masquerade as true V to actively steal the secret of P. How should P and V conduct secure identification through such C ? (See Figure 1.)

This paper considers the problem of how a prover and a verifier securely conduct an identification procedure through an insecure communication channel. Here, the term 'secure' is used in the following two meanings:

- The event that a prover not knowing true secret is accepted by a verifier occurs only with negligible probability. The value of the probability can be set appropriately owing to the strategy in which the verifier discontinues the identification for a prover who has consecutively failed over predetermined times.
- It is computationally infeasible to infer the secret from the observations of interactions between a true prover and a true or fake verifier. Though "computationally" can be replaced by "unconditionally" if so-called one-time password schemes are adopted, but they require huge amount of memory and are impractical for direct human identification to be considered in this paper.

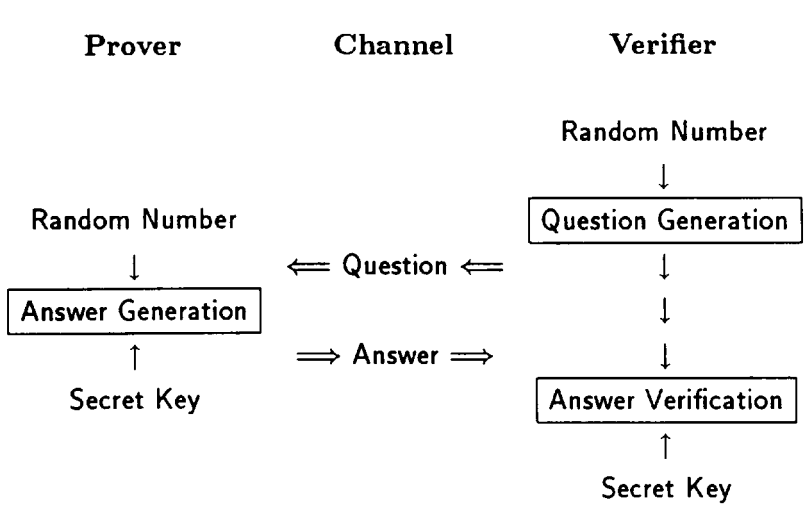

Figure 2. Outline of Interaction

This paper examines a solution to this problem under the following conditions.

- Any attacker cannot directly observe the secret activity concerning the identification inside human brains.
- Any human can easily separate each character of a given sequence into the character she/he remembers and others.
- Any human can memorize a sequence of characters of certain length.
- Any human can select a character randomly from a predetermined small alphabet.
- Unlike conventional so-called dynamic password schemes, no auxiliary devices like pocket calculators are required to assist human provers.

The basic scheme to be proposed is a two-move protocol as illustrated in Figure 2. It is based on common-key cryptography. The verifier asks a random question to the prover. In reply to the question, the human prover generates and sends a random answer based on the common secret key of the prover and the verifier. Finally, the verifier uses the secret key to decide whether the transferred answer matches the question or not. Typical requirement for a human prover is to memorize 10 to 25 characters and to read and input dozens of characters.

The rest of this paper is organized as follows. Section 2. proposes a concrete scheme. Section 3. shows that the scheme is effective. And finally Section 4. concludes this paper.

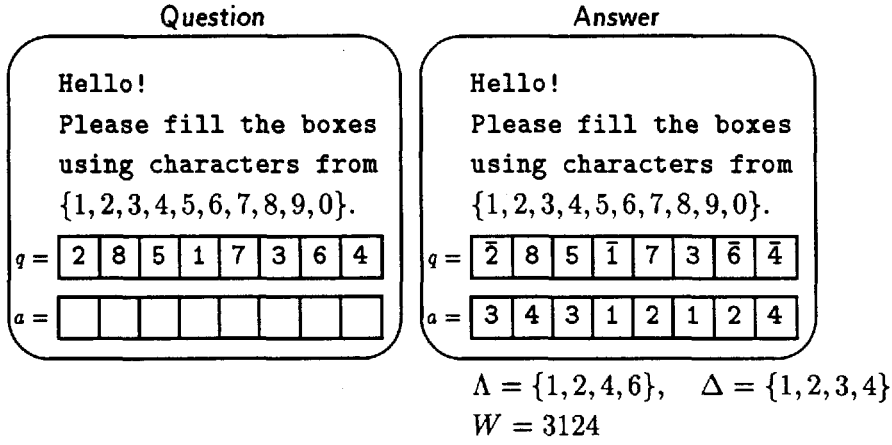

Figure 3. Example of Question and Answer

2 An Identification Scheme

This section proposes a concrete scheme for human identification.

2.1 Demonstration

Figure 3. illustrates a quite simple example of the human identification scheme. In this example, the verifier challenges the prover a question q, a sequence of characters randomly selected from a predetermined alphabet, and requires him to exhibit a sequence a on the alphabet $\{1,2,3,4,5,6,7,8,9,0\}(=\Omega)$ as an answer to q. Ω is called the whole alphabet.

The correct prover remembers a secret key consisting of a window alphabet $\Lambda = \{1,2,4,6\}$, a secret word $W = 3124$, and an answer alphabet $\Delta = \{1,2,3,4\}(\subset \Omega)$.

The correct answer a matching the question q is defined as follows. A hidden window f is embedded in q. The window is a sequence of positions where the characters belonging to Λ are located. At the positions in the answer a corresponding to those in the window f, the characters in the secret word W are located in order. That is, the first character of W (i.e., 3) is assigned to the first position of f (where 2 lies in q), the second character of W (i.e., 1) is assigned to the second position of f (where 1 lies in q), and so on. And at the other positions in a, characters randomly chosen from the answer alphabet Δ are assigned.

Since the prover can separate the characters in Λ from others, he can extract the window f. Or, intuitively, only the correct prover can see the bars marked over the characters in q. Accordingly he can construct the correct answer a by using f and W and Δ. However, even an attacker who has observed a matched question–answer pair cannot be advantageous in correctly answering another question.

Consequently, such a simple trick provides an effective way of enhancing the security of human identification schemes against the insecure environment.

2.2 Notation and Definition

To rigorously describe the identification protocol, the above notions like 'questions', 'answers', and 'windows' should be redefined in terms of functions.

Notation

- The number of elements of a finite set S is denoted by $\#S$.
- The set of all positive integers less than or equal to n is denoted by $\langle n \rangle$.
- The set of all functions (surjections, bijections, resp.) from a set S into a set T is denoted by $\mathrm{Ft}(S,T)$ ($\mathrm{St}(S,T)$, $\mathrm{Bt}(S,T)$, resp.).
- The composite function of functions f and g is denoted by $g \circ f$.

Definition 1. For a totally ordered finite set $(S, \leq)$, let define a function $\mathrm{sort}(S)$ as the bijection $b \in \mathrm{Bt}(\langle \#S \rangle, S)$ such that

$$b(1) \leq b(2) \leq \cdots \leq b(\#S).$$

2.3 Protocol Description

Preparation

A prover P and a verifier V agree on integers ω, γ, λ, δ, β, α, and sets Ω, Γ, Λ, Δ, and a function W, defined in Table 1. See also Figure 4. Among these, Λ, Δ, W constitute a secret key between P and V, while ω, γ, λ, δ, β, α, Ω, Γ are not required to be kept secret.

Table 1. Objects

symbol	name	definition	unit
ω	the whole alphabet size		characters
γ	the question alphabet size	integers satisfying	characters
λ	the window size	$2 \leq \delta \leq \lambda < \gamma \leq \omega$	characters
δ	the answer alphabet size		characters
β	the number of blocks	integers satisfying	
α	the threshold	$1 \leq \alpha \leq \beta$	
Ω	the whole alphabet	a finite set of characters	
Γ	the question alphabet	a subset of Ω with $\#\Gamma = \gamma$	
Λ	a window alphabet	a subset of Γ with $\#\Lambda = \lambda$	
Δ	an answer alphabet	a subset of Ω with $\#\Delta = \delta$	
W	a secret word	a surjection from $\langle\lambda\rangle$ onto Δ	
q_j	a question block	a bijection from $\langle\gamma\rangle$ onto Γ	
Q	a question	$Q = [q_1, \ldots, q_\beta]$	
f_j	a window	an injection from $\langle\lambda\rangle$ into $\langle\gamma\rangle$ such that $f_j = \mathrm{sort}\{i \in \langle\gamma\rangle \mid q_j(i) \in \Lambda\}$	
F	a window tuple	$F = [f_1, \ldots, f_\beta]$	
J	a sieve	a subset of $\langle\beta\rangle$ with $\#J = \alpha$	
a_j	an answer block	a surjection from $\langle\gamma\rangle$ onto Δ such that $a_j \circ f_j = W$ iff $j \in J$	
A	an answer	$A = [a_1, \ldots, a_\beta]$	
D	a decision	a predicate such that $D =$ **accept** if $\#\{j \in \langle\beta\rangle \mid a_j \circ f_j = W\} \geq \alpha$; $D =$ **reject** otherwise	
ν	the question size (the answer size)	$\nu = \beta \cdot \gamma$	characters
μ	an upperbound of required human memory	$\mu = 2\lambda$	characters

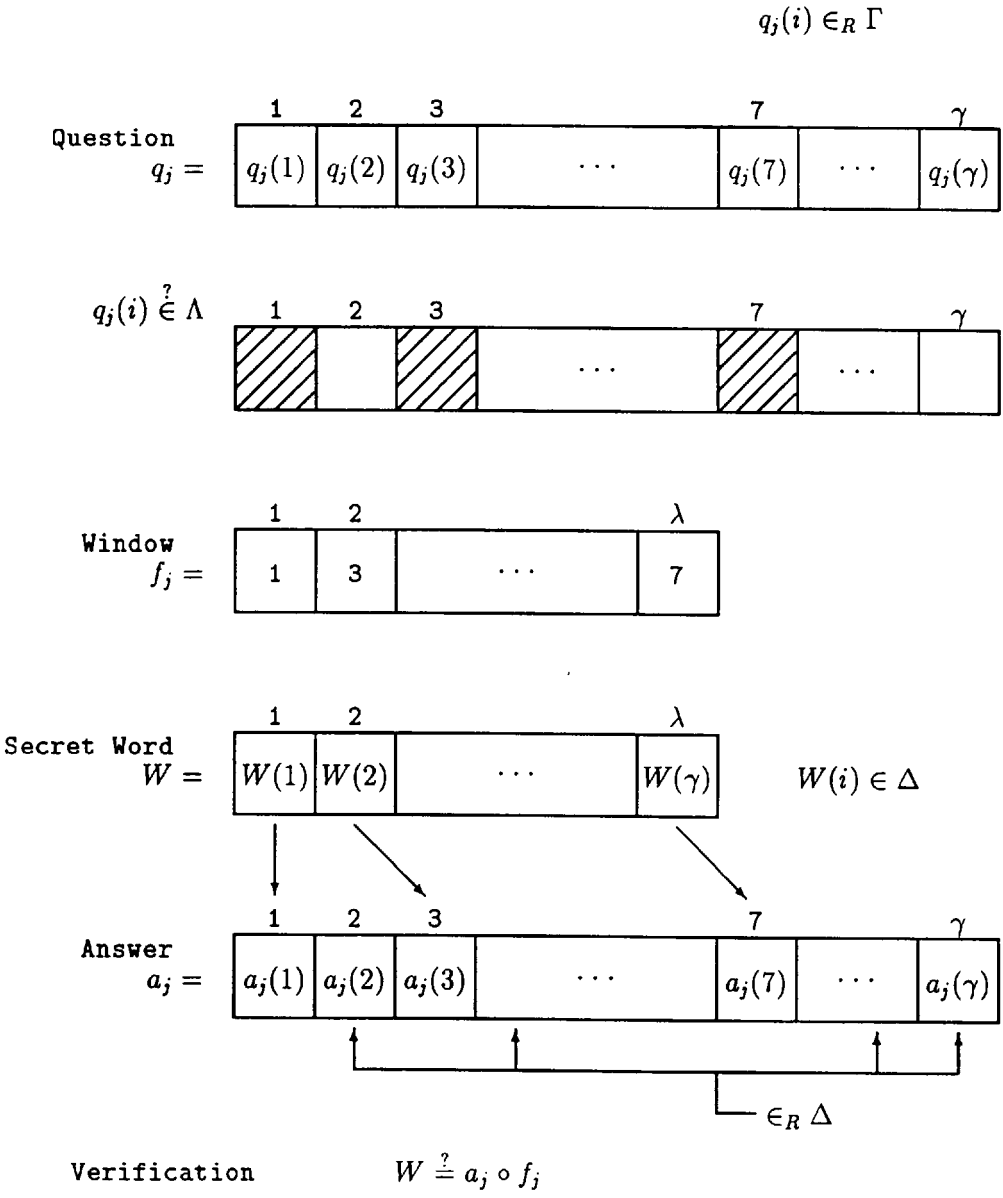

Figure 4. Illustration of Protocol

Interaction (see Figure 4 and Table 1.)

1. **Question Generation:** V selects β question blocks $q_1, \ldots, q_\beta$ randomly and uniformly from $\mathsf{Bt}(\langle\gamma\rangle, \Gamma)$.

2. **Question Transfer:** V transfers the question $Q = [q_1, \ldots, q_\beta]$ to P.

3. **Answer Generation:** P generates β answer blocks $a_1, \ldots, a_\beta \in \mathsf{St}(\langle\gamma\rangle, \Delta)$ as follows:

   3-1. Sieve Selection: P selects α distinct elements randomly and uniformly from $\langle\beta\rangle$ to form a sieve $J \subseteq \langle\beta\rangle$.

   3-2. Correct Answer Block Generation: P generates answer block a_j for each $j \in J$ as follows:

   3-2-1. Question Block Check: If $q_j \in \mathsf{Bt}(\langle\gamma\rangle, \Gamma)$ then P proceeds. Otherwise P quits.

   3-2-2. Window Detection: P computes the window $f_j = \mathsf{sort}(\{i \in \langle\gamma\rangle | q_j(i) \in \Lambda\}) \in \mathsf{Ft}(\langle\lambda\rangle, \langle\gamma\rangle)$.

   3-2-3. Secret Word Embedding: For each $i \in f_j(\langle\lambda\rangle) = \{i \in \langle\gamma\rangle | \exists h \in \langle\lambda\rangle, i = f_j(h)\}$, P determines $a_j(i)$ as $a_j(i) = W(h)$.

   3-2-4. Random Padding: For each $i \in \langle\gamma\rangle - f_j(\langle\lambda\rangle)$, P selects an element randomly and uniformly from Δ and allocates it to $a_j(i)$.

   3-3. Random Answer Block Generation: For each $j \in \langle\beta\rangle - J$, P selects a surjection randomly and uniformly from $\mathsf{St}(\langle\gamma\rangle, \Delta)$ and allocates it to a_j.

4. **Answer Transfer:** P transfers the answer $A = [a_1, \ldots, a_\beta]$ to V.

5. **Answer Verification:** V evaluates the decision D as follows:

   5-1. Window Detection: For each $j \in \langle\beta\rangle$, V computes the window $f_j = \mathsf{sort}(\{i \in \langle\gamma\rangle | q_j(i) \in \Lambda\}) \in \mathsf{Ft}(\langle\lambda\rangle, \langle\gamma\rangle)$ to form $F = [f_1, \ldots, f_\beta]$.

   5-2. Embedded Secret Word Check: If $\#\{j \in \langle\beta\rangle | a_j \circ f_j = W\} \geq \alpha$ then V puts $D =$ **accept** and accepts P. Otherwise V puts $D =$ **reject** and rejects P.

Remark: V can do **5-1.** at any time after **1.** and before **5-2.**

3 Evaluation

This section shows the effectiveness of the proposed identification scheme.

3.1 Logical Completeness

Proposition 1. In the proposed scheme, any prover knowing correct Λ, Δ, W and obeying the protocol is certainly accepted by the verifier.

3.2 Tolerance against Random Attacks

The followings are some typical attacks done without secret keys.

Definition 2.

A known-Δ random attack means the following attack conducted by an attacker who knows the protocol itself and knows the concrete values of ω, γ, λ, δ, Ω, Γ, and Δ, but doesn't know concrete values of Λ and W. In reply to any question Q the attacker transfers an answer $A = [a_1, \ldots, a_\beta]$, each block a_j of which is selected randomly and uniformly from $\mathrm{St}(\langle\gamma\rangle, \Delta)$.

The success probability of a known-Δ random attack is the probability p_Δ that a question Q and an answer A by a known-Δ random attack satisfy $D =$ **accept**.

A Δ-guessing random attack means the following attack conducted by an attacker who knows the protocol itself and knows the concrete values of ω, γ, λ, δ, Ω, and Γ, but doesn't know concrete values of Λ, Δ, and W. In reply to any question Q, the attacker firstly guesses Δ as a subset of Ω with δ elements, then transfers an answer $A = [a_1, \ldots, a_\beta]$, each block a_j of which is selected randomly and uniformly from $\mathrm{St}(\langle\gamma\rangle, \Delta)$.

The success probability of a Δ-guessing random attack is the probability p_δ that a question Q and an answer A by a Δ-guessing random attack satisfy $D =$ **accept**.

By straightforward calculation the success probability of each attacks can be estimated as follows.

Proposition 2. The success probability of a known-Δ attack is given by

$$p_\Delta = \sum_{j=\alpha}^{\beta} \binom{\beta}{j} p^j (1-p)^{\beta-j} \leq \binom{\beta}{\max\{\alpha, \lfloor \beta/2 \rfloor\}} p^\alpha,$$

where

$$p = \frac{1}{\delta^\lambda (1 - \sum_{i=1}^{\delta-1} \binom{\delta}{i} (\frac{i}{\delta})^\gamma)}.$$

And the success probability of a Δ-guessing random attack is given by

$$p_\delta = \frac{p_\Delta}{\binom{\omega}{\delta}}.$$

In summary, an attacker E not knowing Δ but knowing δ only can do a Δ-guessing random attack which succeeds with probability p_δ. But, if E can observe a question Q and an answer A transferred between a verifier V and a prover P who is accepted by V, then E can obtain Δ and can do a known-Δ random attack with success probability $p_\Delta = \binom{\omega}{\delta} \cdot p_\delta$. An interesting problem arising here is to look for more powerful attacks which can be done by such an attacker E observing interactions of P and V.

3.3 Complexity of Inferring Secret

Let's analize the case $\alpha = \beta = 1$. Similar results hold for the general case.

Definition 3. For any $\Lambda \subset \Gamma$ and any bijection $q : \langle\gamma\rangle \longrightarrow \Gamma$, let define an injection

$$f_{\Lambda,q} = \mathrm{sort}(\{i \in \langle\gamma\rangle | q(i) \in \Lambda\}),$$

and call it the window determined by a window alphabet Λ and a question (block) q.

Definition 4. For any $\Lambda \subset \Gamma$ and any surjection $W : \langle\lambda\rangle \longrightarrow \Delta$, let define a set $C_{\Lambda,W}$ by

$$C_{\Lambda,W} = \{(q,a) | q : \langle\gamma\rangle \longrightarrow \Gamma, a : \langle\gamma\rangle \longrightarrow \Delta, a \circ f_{\Lambda,q} = W\}.$$

Also define a set C by

$$C = \{C_{\Lambda,W} | \Lambda \subset \Gamma, W : \langle\lambda\rangle \longrightarrow \Delta\}.$$

An attacker can obtain a set O such that $O \subseteq C_{\Lambda,W} \in C$ for unknown Λ and W, by observing correct interactions between a prover and a verifier, or even by masquerading as a verifier to collect answers from a prover. Such a set O is called an observation.

Definition 5. Let B denote the set of all observations, *i.e.*,

$$B = \{O | \exists C_{\Lambda,W} \in C \quad \text{such that} \quad O \subseteq C_{\Lambda,W}\}.$$

A target of an attacker given an observation $O \in B$ is to determine $\Lambda \subset \Gamma$ and $W : \langle\lambda\rangle \longrightarrow \Delta$ such that $O \subseteq C_{\Lambda,W}$.

It is easy to see the following proposition holds.

Proposition 3. For any observation $O \in B$, there exists a window alphabet $\Lambda(\subset \Gamma)$ such that

$$a \circ f_{\Lambda,q} = a' \circ f_{\Lambda,q'}$$

for any (q, a), $(q', a') \in O$.

Accordingly, the following strategy of attack can be derived.

Strategy. Since an attacker given O can check for each $\Lambda(\subset \Gamma)$ with $\#\Lambda = \lambda$ whether the condition in Proposition 3 holds or not, the attacker can find out correct Λ in at most $\begin{pmatrix} \gamma \\ \lambda \end{pmatrix}$ trials. From this correct Λ, the attacker can obtain the secret word W as $W = a \circ f_{\Lambda,q}$.

3.4 Viability

Let's consider the following examples.

Example 1. If $\omega = \gamma = 36$, $\lambda = 18$, $\delta = 2$, and $\beta = \alpha = 1$, then

$$p_\delta \simeq 6.06 \times 10^{-9},$$
$$p_\Delta \simeq 3.81 \times 10^{-6},$$
$$\begin{pmatrix} \gamma \\ \lambda \end{pmatrix} \simeq 9.08 \times 10^9 > 2^{33},$$

and $\nu = 36$ and the required human memory is at most 18 characters + 5 hexadecimal digits.

Example 2. If $\omega = \gamma = 50$, $\lambda = 10$, $\delta = 3$, and $\beta = \alpha = 1$, then

$$p_\delta \simeq 8.64 \times 10^{-10},$$
$$p_\Delta \simeq 1.69 \times 10^{-6},$$
$$\binom{\gamma}{\lambda} \simeq 1.03 \times 10^{10},$$

and $\nu = 50$ and the required human memory is at most 10 characters + 5 nonary digits.

From these examples, with memory, computational and communication complexity acceptable for a human prover and even for a human verifier, the proposed scheme can keep enough security over human identification through insecure channel.

4 Conclusion

This paper has described the importance of the problem of human identification through insecure channels, and proposed a concrete scheme which is matching to human ability of memorizing and processing. The following items are listed for further research :

- Applying the proposed scheme into practice.
- Developing much simpler (lower complexity) versions.
- Studying simple schemes for a human to read a message from an entity through an insecure channel and to reply a confirmation of receiving it through the channel.
- Studying the case where not the both but one of the channel from which a human receives messages and the channel to which a human sends messages is insecure.

Acknowledgment

It is a pleasure to thank Steve Babbage, Donald Davies, Hiroyuki Masumoto, James Massey, Hiroshi Miyano, Satoshi Ozaki, Yuichi Saitoh, Minoru Sasaki, Taroh Sasaki, Kazue Tanaka, Yacov Yacobi, Hiroharu Yoshikawa for their interest and comments.

Reference

[1] D. W. Davies and W. L. Price, *Security for Computer Networks*, Chapter 7, John Wiley & Sons, 1984.